### Recap

Major Activities of Requirements Engineering: Elicitation, Analysis, Specification, Validation. This week focus on analysis

Recap: Types of Requirements

* Functional: describe functions
  + E.g.: SW enable student enroll in class
* non-functional: requirement that constrain solution
  + performance: sw respond client web in a timely and convenient way
  + maintainability: SW implement using modern programming practices maximize maintainability and reusability of code.
  + Safety: SW will transition to an agreed safe state within 1 second of any sensor readings outside their thresholds
  + Reliability: SW shall be available for use as much as comparable productivity tools.
  + Security: SW shall protect users’ personal information from XXX penetration attacks
  + Privacy: SW shall protect each user account with password entry
  + Interoperability: SW shall run on XX phones and YY devices
  + Useability: SW shall conform to ISO 9241 usability standard
  + Student addresses and other information should not released by unauthorized party
* More classifications
  + Priority: Higer priority means more essential requirement meet goal of SW. Fixed point scale: Must have, Should have, Could have, and Won’t have
  + Scope: which requirement affect SW and component. E.g.: repond time too low effect satisfaction
  + Volatility: requirement change during lifetime of SW during development. estimate the likelihood requirement will change, so developer can tolerant of change

Viewpoints of Requirements

* Interactor: people or other system interact with system
* Indirect: Stakeholder who don’t use system but effect requirement: payroll system
* Domain: constraint of domain effect requirement

### Requirement Negotiation or Conflict Resolution

Both resolve problem with requirement where conflict happen

* Between requirements and resources
* between functional and non-functional
* between stakeholders

Don’t make unilateral decision, consult stakeholder, contractual reasons, traceable back to customer

### Detecting Conflicts

Priority:

* importance to customer
* time and resources
* conflicting requirement
* future releases: which done first, which left to future

MoSCoW method (Must have, Should have, Could have)

* ask client group requirement to DO list and NOT DO
* construct mathematical model

Formal Methods

* use logical analysis verify properties and identify inconsistencies
* most method have tool support with automatic analysis
* Weakness
  + Client don’t clear with their priorities.
  + Different stakeholders don’t agree priority list
  + Requirement are NOT independent, all strategies assume they are

### Resolving Conflicts

Why negotiation

* Aid understanding and encourage communication
* Reveal conflict, solution, collaborative resolution
* Improve agreement level and requirement quality and stakeholder satisfaction

Negotiation for agile software development

* Focus on reveal conflict, understand requirement
* Agile method focus on involvement of customer